Protection Schemes for Two Link Failures in Optical Networks

Salah A. Aly and Ahmed E. Kamal
Department of Electrical and Computer Engineering
Iowa State University, Ames, IA 50011, USA
Email: {salah,kamal}@iastate.edu

November 12, 2008

***Abstract*—In this paper we develop network protection schemes against two link failures in optical networks. The motivation behind this work is the fact that the majority of all available links in an optical network suffer from single and double link failures. In the proposed network protection schemes, NPS2-I and NPS2-II, we deploy network coding and reduced capacity on the working paths to provide backup protection paths. In addition, we demonstrate the encoding and decoding aspects of the proposed schemes.**

***Index Terms*—Network Protection, Optical Networks.**

I. Introduction

One of the main services of operation networks that must be deployed efficiently is reliability. In order to deploy a reliable networking strategy, one needs to protect the transmitted signals over unreliable links. Link failures are common problems that might occur frequently in single and multiple operating communication circuits. In network survivability and network resilience, one needs to design efficient strategies to overcome this dilemma. Optical network survivability techniques are classified as *pre-designed protection* and *dynamic restoration* [13], [8]. The approach of using pre-designed protection aims to reserve enough bandwidth such that when a failure occurs, backup paths are used to reroute the transmission and be able to recover the data. Examples of this type are 1-1 and 1-N protections [2], [9]. In dynamic restoration reactive strategies, capacity is not reserved. However, when the failure occurs, dynamic recovery is used to recover the data transmitted in the links that are suffered from failures. This technique does not need preserved resources or provision of extra paths that work in case of failure. In this work we will provide several strategies of dynamic restoration based on coding and reduced distributed fairness capacities.

Network coding is a powerful tool that has been recently used to increase the throughput, capacity, and performance of communication networks. Information theoretic aspects of network coding have been investigated in [12], [1]. Network coding allows the intermediate nodes not only to forward packets using network scheduling algorithms, but also encode/decode them using algebraic primitive operations, see [1], [4], [12] and references therein. As an application of network coding, data loss because of failures in communication links can be detected and recovered if the sources are allowed to perform network coding operations. Network coding is used to maximize the throughput [1], [10]. Also, it is robust against packet losses and network failures [5], [11], [6].

Network protection against single and multiple link failures using adding extra protection paths has been introduced in [7], [8]. The source nodes are able to combine their data into extra paths (backup *protection paths*) that are used to protect all signals on the *working paths* plain carrying data from all sources. In both cases, p-cycles has been used for protection against single and multiple link failures.

In this paper we design two schemes for network protection against one and two failed links in a network with n disjoint *working paths*: NPS2-I and NPS2-II. The approach is based on network coding data originated by the sources. We assume that network capacity will be reduced by a partial factor in order to achieve the required protection. Several advantages of NPS2-I and NPS2-II can be stated as:

- The data sent from the sources are protected without adding extra paths. Two paths out of the n disjoint *working paths* will carry encoded data, and hence they are *protection paths*.
- The encoding and decoding operations are achieved with less computational cost at both the sources and receivers. The recovery from failures is achieved immediately without asking the senders to retransmit the lost data.
- The normalized network capacity is $(n-2)/n$, which is near-optimal in case of using a large number of connections.

II. Network Model

In this section we present the network model and basic terminology.

i) Let $\mathcal{N}$ be a network represented by an abstract graph $G=(\mathbf{V},E)$, where $\mathbf{V}$ is a set of nodes and E be a set of undirected edges. Let S and R be sets of independent sources and destinations, respectively. The set $\mathbf{V}=V\cup S\cup R$ contains the relay, source, and destination nodes. Assume for simplicity that $|S|=|R|=n$, hence the set of sources is equal to the set of receivers.

ii) A path (connection) is a set of edges connected together with a starting node (sender) and an ending node (receiver).

$$L_i=\{(s_i,e_{1i}),(e_{1i},e_{2i}),\ldots,(e_{(m)i},r_i)\},$$

where $1 \leq i \leq n$ and $(e_{(j-1)i}, e_{ji}) \in E$ for some integer m.

iii) The node can be a router, switch, or an end terminal depending on the network model $\mathcal{N}$ and the transmission layer.

iv) L is a set of paths $L = \{L_1, L_2, \ldots, L_n\}$ carrying the data from the sources to the receivers as shown in Fig. 2. All connections have the same bandwidth, otherwise a connection with high bandwidth can be divided into multiple connections, each of which has a unit capacity. There are exactly n connections. For simplicity, we assume that the number of sources is less than or equal to the number of available paths. A sender with a high capacity can divide its capacity into multiple unit capacities, each of which has its own path. The failure on a link L_i may occur due to the network circumstance such as a link replacement, overhead, etc. We do not address in this work the failure cause. However, we assume that there are one or two failures in the working paths and the protection strategy is able to protect/recover it.

v) Each sender $s_i \in S$ will transmit its own data x_i to a receiver r_i through a connection L_i. Also, s_i will transmit encoded data $\sum_i^n x_i$ to r_i at a different time slot if it is assigned to send the encoded data.

vi) The data from all sources are sent in sessions. Each session has a number of time slots n. Hence t_δ^ℓ is a value at round time slot ℓ in session δ.

vii) In this model $\mathcal{N}$ if we consider only a single link failure, then it is sufficient to apply the encoding and decoding operations over a finite field with two elements, $\mathbf{F}_2 = \{0, 1\}$. However, if there are double failures, then a finite field with higher alphabets is required.

viii) There are at least two receivers and two senders with at least two disjoint paths. Otherwise, the protection strategy can not be deployed for a single path, which it can not protect itself.

We can define the network capacity in the light of min-cut max-flow information theoretic view [1].

***Definition** 1:* The capacity of a connecting path L_i between s_i and r_i is defined by

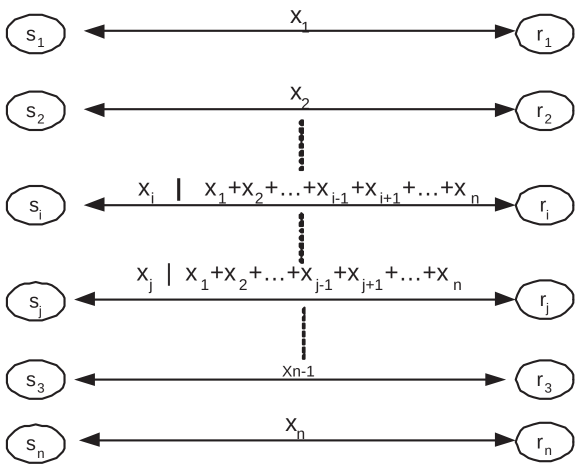
$$c_i = \begin{cases} 1, & L_i \text{ is } \textit{active}; \\ 0, & \text{otherwise}. \end{cases} \tag{1}$$

The total capacity is given by the summation of all paths' capacities. What we mean by an *active* link is that the receiver is able to receive and process signals/messages throughout this link.

Clearly, if all links are active then the total capacity is n and normalized capacity is 1. In general the normalized capacity of the network for the active and failed links is computed as:

$$C_{\mathcal{N}} = \frac{1}{n} \sum_{i=1}^{n} c_i. \tag{2}$$

We define the *working paths* and *protection paths* as follows:



s_1
x_1
r_1
s_2
x_2
r_2
s_i
$x_i \mid x_1+x_2+\ldots+x_{i-1}+x_{i+1}+\ldots+x_n$
r_i
s_j
$x_j \mid x_1+x_2+\ldots+x_{j-1}+x_{j+1}+\ldots+x_n$
r_j
s_3
Xn-1
r_3
s_n
x_n
r_n

Fig. 1. Network protection against link/path failures using reduced capacity and network coding. Two paths out of n disjoint *working paths* carry encoded data for protection against two link failures.

***Definition** 2:* The *working paths* of a network with n connection paths are the paths that carry unencoded data traffic under normal operations. The *Protection paths* are paths that provide alternate backup paths to carry encoded data traffic in case of failures. A protection scheme ensures that data sent from the sources will reach the receivers in case of failure incidences on the working paths.

Every sender s_i prepares a packet $packet_{s_i \rightarrow r_i}$ to send to the receiver r_i. The packet contains the sender's ID, data x_i^ℓ, and a round time for every session t_δ^ℓ for some integers δ and ℓ. We have two types of packets:

i) Packets sent without coding, in which the sender does not need to perform any coding operations. For example, in case of packets sent without coding, the sender s_i sends the following packet to the receiver r_i.

$$packet_{s_i \rightarrow r_i} := (ID_{s_i}, x_i^\ell, t_\delta^\ell) \tag{3}$$

ii) Packets sent with encoded data, in which the sender needs to send other sender's data. In this case, the sender s_i sends the following packet to receiver r_i:

$$packet_{s_i \rightarrow r_i} := (ID_{s_i}, \sum_{j=1, j \neq i}^{n} x_j^\ell, t_\delta^\ell). \tag{4}$$

The value $y_i^\ell = \sum_{j=1, j \neq i}^{n} x_j^\ell$ is computed by every sender s_i, by collecting the data from all other senders and encoding them using the bit-wise operation.

In either case, the sender has a full capacity in the connection path L_i.

The protection path that carries the encoded data from all sources is used for the data recovery in case of failure. Assuming the encoding operations occur in the same round time of a particular session, every source s_i adds its value, for $1 \leq i \leq n$. Therefore, the encoded data over the protection path is $y_i = \sum_{j=1, i \neq j}^{n} x_j$. The decoding operations are done at every receiver r_i by adding the data x_i received over the working path L_i. The node r_k with failed connection L_k will be able to recover the data x_k. Assuming all operations are achieved over the binary finite field $\mathbf{F}_2$. Hence we have

$$x_k = y_i - \sum_{j=1, i \neq j}^{n} x_j. \tag{5}$$

III. Protections Using Dedicated Paths (NPS2-I)

In this section we develop a network protection scheme (NPS2-I) for two link failures in optical networks. The protection scheme is achieved using network coding and dedicated paths. Assume we have n connections carrying data from a set of n sources to a set of n receivers. All connections represent disjoint paths, and the sources are independent of each other. The authors in [7], [3] introduced a model for optical network protection against a single link failure using an extra and dedicated paths provisioning. In this model NPS2-I we extend this approach to two link failures.

We will provide two backup paths to protect against any two disjoint links, which might experience failures. These two protection paths can be chosen using network provisioning. The protection paths are fixed for all rounds per session, but they may vary among sessions. For example, sender s_i transmits a message x_i^ℓ to a receiver r_i at time t_δ^ℓ in round time ℓ in session δ. This process is explained in Scheme (6) as:

	round time session 1					$\dots$
	1	2	3	$\dots$	n	$\dots$
$s_1 \rightarrow r_1$	x_1^1	x_1^2	x_1^3	$\dots$	x_1^n	$\dots$
$s_2 \rightarrow r_2$	x_2^1	x_2^2	x_2^3	$\dots$	x_2^n	$\dots$
$s_3 \rightarrow r_3$	x_3^1	x_3^2	x_3^3	$\dots$	x_3^n	$\dots$
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\dots$
$s_i \rightarrow r_i$	x_i^1	x_i^2	x_i^3	$\dots$	x_i^n	$\dots$
$s_j \rightarrow r_j$	y_j^1	y_j^2	y_j^3	$\dots$	y_j^n	$\dots$
$s_k \rightarrow r_k$	y_k^1	y_k^2	y_k^3	$\dots$	y_k^n	$\dots$
$s_{i+1} \rightarrow r_{i+1}$	x_{i+1}^1	x_{i+1}^2	x_{i+1}^3	$\dots$	x_{i+1}^n	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\dots$
$s_n \rightarrow r_n$	x_n^1	x_n^2	x_n^3	$\dots$	x_n^n	$\dots$

(6)

All y_j^ℓ's are defined as:

$$y_j^\ell = \sum_{i=1, i \neq j \neq k}^{n} a_i^\ell x_i^\ell \text{ and } y_k^\ell = \sum_{i=1, i \neq k \neq j}^{n} b_i^\ell x_i^\ell. \tag{7}$$

The coefficients a_i^ℓ and b_i^ℓ are chosen over a finite field $\mathbf{F}_q$ with $q > n-2$, see [3] for more details. One way to choose these coefficients is by using the follow two vectors.

$$\begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \alpha & \alpha^2 & \dots & \alpha^{n-3} \end{bmatrix} \tag{8}$$

Therefore, the coded data is

$$y_j^\ell = \sum_{i=1, i \neq j \neq k}^{n} x_i^\ell \text{ and } y_k^\ell = \sum_{i=1, i \neq k \neq j}^{n} \alpha^{i \mod n-2} x_i^\ell. \tag{9}$$

In the case of two failures, the receivers will be able to solve two linearly independent equations in two unknown variables. For instance, assume the two failures occur in paths number two and four. Then the receivers will be able to construct two equations with cofficients

$$\begin{bmatrix} 1 & 1 \\ \alpha & \alpha^3 \end{bmatrix} \tag{10}$$

Therefore, we have

$$x_2^\ell + x_4^\ell \tag{11}$$

$$\alpha x_2^\ell + \alpha^3 x_4^\ell \tag{12}$$

One can multiply the first equation by α and subtract the two equations to obtain value of x_4^ℓ.

We notice that the encoded data symbols y_j^ℓ and y_k^ℓ are fixed per one session transmission but it is varied for other sessions. This means that the path L_j is dedicated to send all encoded data $y_j^1, y_j^2, \dots, y_j^n$.

***Lemma** 3:* The normalized capacity of NPS2-I of the network model $\mathcal{N}$ described in (6) is given by

$$\mathcal{C} = (n-2)/n. \tag{13}$$

Proof: There are n rounds in every session. Also, we have n connections per a round time. There exist two connections which carry backup data for protection, hence there are $n-2$ connections that carry working data. Therefore, the normalized capacity is given as:

$$\mathcal{C} = (n-2)n/n^2,$$

which gives the result. ■

In NPS2-I there are three different scenarios for two link failures, which can be described as follows:

i) If the two link failures occur in the backup protection paths L_j and L_k, then no recovery operations are required at the receivers side. The reason is that these two paths are used for protections, and all other working paths will convey the data from the senders to receivers.

ii) If the two link failures occur in one backup protection path say L_j and one working path L_i, then recovery operations are required. The receiver r_i must recover its data using one of the protection paths.

iii) If the two link failures occur in two working paths, then in this case the two protection paths are used to recover the lost data. The idea of recovery in this case is to build a system of two equations with two unknown variables.

IV. Protection Against Two Link Failures (NPS2-II)

In this section we will provide an approach for network protection against two link failures in optical networks. We deploy network coding and distribute capacity over the *working paths*. We will compute the network capacity in this approach. In [3] we will illustrate the tradeoff and implementation aspects of this approach, where there is enough space for details.

We assume that there is a set of n connections that need to be protected with $\%100$ guarantee against single and two link failures. Assume $\mathbf{F}_q$ is a finite field with q elements. For simplicity we consider n is an even number.

A. *Two Link Failures, Achieving $(n-2)/n$ Capacity*

Let x_i^ℓ be the data sent from the source s_i at round time ℓ in a session at time t_δ^ℓ. Also, assume the encoded data $y_i = \sum_{j=1, j\neq i}^{n} x_j^\ell$. Put differently:

$$y_i = x_1^\ell \oplus x_2^\ell \oplus \ldots \oplus x_{i\neq j}^\ell \oplus \ldots \oplus x_n^\ell. \tag{14}$$

The protection scheme NPS2-II runs in sessions as explained below. Every session has at most one single failure through out its each round time. As shown in Scheme (15), the protection matrix for the first session is given by the following protection code:

	round time session 1					
	1	2	3	$\ldots$	$\ldots$	ℓ
$s_1 \longrightarrow r_1$	y_1^1	x_1^1	x_1^2	$\ldots$	$\ldots$	$x_1^{\ell-1}$
$s_2 \longrightarrow r_2$	y_2^1	x_2^1	x_2^2	$\ldots$	$\ldots$	$x_2^{\ell-1}$
$s_3 \longrightarrow r_3$	x_3^1	y_3^2	x_3^2	$\ldots$	$\ldots$	$x_3^{\ell-1}$
$s_4 \longrightarrow r_4$	x_4^1	y_4^2	x_4^2	$\ldots$	$\ldots$	$x_3^{\ell-1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$s_i \longrightarrow r_i$	x_i^1	x_i^2	$\ldots$	y_i^j	$\ldots$	$x_i^{\ell-1}$
$s_{i+1} \longrightarrow r_{i+1}$	x_{i+1}^1	x_{i+1}^2	$\ldots$	y_{i+1}^j	$\ldots$	$x_{i+1}^{\ell-1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$s_{n-1} \longrightarrow r_{n-1}$	x_{n-1}^1	x_{n-1}^2	x_{n-1}^3	$\ldots$	$\ldots$	y_{n-1}^ℓ
$s_n \longrightarrow r_n$	x_n^1	x_n^2	x_n^3	$\ldots$	$\ldots$	y_n^ℓ

(15)

where

$$y_k^\ell = \sum_{i=1}^{2(\ell-1)} a_i^{\ell-1} x_i^{\ell-1} + \sum_{i=2\ell+1}^{n} a_i^\ell x_i^\ell$$
$$\text{for } (2\ell-1) \leq k \leq 2\ell, 1 \leq \ell \leq n/2. \tag{16}$$

All coefficients are taken from $\mathbf{F}_q$ for $q > n-2$, see [3] for more details. Also, the two vectors shown in 8 can be used in this case. We note that the data symbols in NPS2-II are sent in independent sessions. This means that every session has its own recovery scheme. Also, two failures occur in only two connections during the session round times. Hence the sender s_i sends the message x_i^j for all $1 \leq j \leq \ell - 1$ and $1 \leq i \leq n$ during the first session. One can always change the round time of the encoded data y_k^ℓ and the data x_i^j for any round time j in the same session.

Now, we shall compute the normalized capacity of NPS2-II for the network $\mathcal{N}$ at one particular session; the first session. The capacity is calculated using the well-known min-cut max-flow theorem [1].

***Theorem** 4:* The optimal fairness distributed normalized capacity of NPS2-II shown in Scheme (15) is given by

$$\mathcal{C} = (n-2)/n. \tag{17}$$

Proof: Let n be the number of sources, each of which has a unit capacity in the connection L_i from s_i to r_i. Let j be an index of an arbitrary session that has two link failures. We have n paths that have capacity n. Also, we have $\ell = n/2$ round times, in which each round time has $n-2$ capacity in the working paths. Therefore the total capacity is given by

$$(n-2)(\ell) = (n^2 - 2n)/2. \tag{18}$$

By normalizing this value with the total capacity $n\ell$, then the result follows. ■

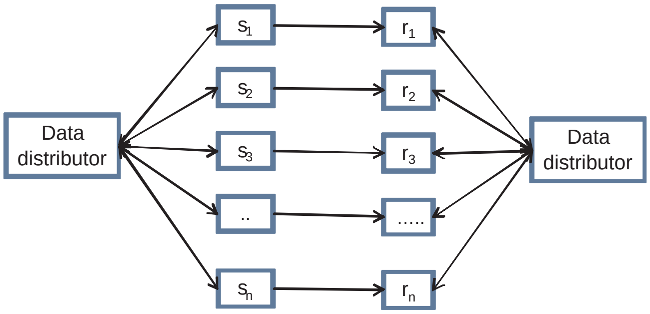


Fig. 2. The encoding and decoding operations are done at the data distributor and collector, respectively.

The network protection strategy NPS2-II against one or two link failures is deployed in two processes: Encoding and decoding operations. The encoding operations are performed at the set of sources, in which one or two sources will send the encoded data depending on the number of failures. The decoding operations are performed at the receivers' side, in which receivers with failed links had to receive all other receivers' data in order to recover their own data. Depending on NPS2-I or NPS2-II the receivers will experience some delay before they can actually decode the packets. The transmission is done in rounds, hence linear combinations of data has to be from the same round time. This can be achieved using the round time that is included in each packet sent by a sender.

Assume there are data collectors $\mathcal{S}$ and $\mathcal{R}$ at the senders and receivers, respectively. They can be a sender (receiver) node to send (receiver) encoded data, see [3].

Encoding Process: The encoding operations are for each round per a session.

- The source nodes send a copy of their data to the data distributor $\mathcal{S}$, then $\mathcal{S}$ will decide which source will send the encoding data y_k^ℓ and all other sources will send their own data x_i^ℓ. This process will happen in every round during transmission time.
- The encoding is done by linear operation of sources' coefficients which is the fastest arithmetic operation that can be performed among all sources' data.
- The server $\mathcal{S}$ will change the sender s_i that should send the encoded data y_i^ℓ in every round of a given session for the purpose of fairness and distributed capacities.

Decoding Process: The objective of the decoding and recovery process is to withhold rerouting the signals or the transmitted packets due to link failures, see [2], [3], [8]. However, we provide strategies that utilize network coding and reduced capacity at the source nodes. We assume that the receiver nodes are able to perform decoding operations using a data collector $\mathcal{R}$.

We assume there is a data distributor $\mathcal{R}$ that will collect data from all *working* and *protection paths* and is able to perform the decoding operations. In this case we assume that all receivers R have available shared paths with the data collector $\mathcal{R}$. At the receivers side, if there are two failures in paths L_j and L_k, then there are several situations.

- If the paths L_j and L_k carry unencoded data (*working paths*), then the data distributor $\mathcal{R}$ must query all other nodes in order to recover the failed data. In this case r_k and r_j must query $\mathcal{R}$ to retrieve their lost data.
- If the path L_k carries encoded data y_k (*protection path*) and L_j carries unencoded data (*working path*), then data collector $\mathcal{R}$ must query all other receivers in order to perform decoding, and r_j receives the lost data x_j^ℓ.
- If the paths L_j and L_k carry encoded data (they are both *protection paths*), then no action is required.

V. Conclusion

In this paper we presented network protection schemes NPS2-I and NPS2-II against single and double link failures in optical networks. We showed that protecting two failures can be achieved using network coding and reduced capacity. The normalized capacity of the proposed schemes is $(n-2)/n$, which is near optimal for a large number of connections. Extended version of this paper will appear in [3], where protection against t multiple failures is investigated.

VI. Acknowledgments

This research was supported in part by grants CNS-0626741 and CNS-0721453 from the National Science Foundation, and a gift from Cisco Systems. *S A. A would like to thank Prof. A E. K for his support and guidance. He is also grateful to Prof. E. Soljanin and Prof. A. Klappenecker for their kindness and for training him how to inscribe high quality research papers.*

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. Network information flow. *IEEE Trans. Inform. Theory*, 46:1204–1216, 2000.

[2] S. A. Aly and A. E. Kamal. Network protection codes against link failures using network coding. In *Proc. IEEE GlobelComm '08, New Orleans, LA*, December 2008.

[3] S. A. Aly and A. E. Kamal. On the construction of network protection codes against network failures. *IEEE Journal on Selected Areas in Communciation (JSAC) , on submission*, 2008.

[4] C. Fragouli, J. Le Boudec, and J. Widmer. Network coding: An instant primer. *ACM SIGCOMM Computer Communication Review*, 36(1):63–68, 2006.

[5] T. Ho, R. Koetter, M. Medard, D. Karger, and M. Effros. The benefits of coding over routing in a randomized setting. In *Proc. of the IEEE International Symposium on Information Theory (ISIT03)*, page 442, Yokohama, Japan, June 2003.

[6] S. Jaggi, M. Langberg, S. Katti, T. Ho, D. Katabi, and M. Medard. Resilient network coding in the presence of byzantine adversaries. In *Proc. IEEE INFOCOM*, 2007.

[7] A. E. Kamal. 1+N protection in optical mesh networks using network coding on p-cycles. In *Proc. of the IEEE Globecom*, 2006.

[8] A. E. Kamal. 1+N network protection for mesh networks: network coding-based protection using p-cycles. *submitted to IEEE Journal of Communication*, 2008.

[9] A. E. Kamal. A generalized strategy for 1+N protection. In *Proc. of the IEEE International Conference on Communications (ICC)*, 2008.

[10] R. Koetter and M. Medard. An algebraic approach to network coding. *IEEE/ACM transactions on networking*, 2003.

[11] D. S. Lun, N. Ranakar, R. Koetter, M. Medard, E. Ahmed, and H. Lee. Achieving minimum-cost multicast: A decentralized approach based on network coding. In *In Proc. the 24th IEEE INFOCOM*, volume 3, pages 1607– 1617, March 2005.

[12] E. Soljanin and C. Fragouli. Network codinginformation flow perspective. 2007.

[13] D. Zhou and S. Subramaniam. Survivability in optical networks. *IEEE network*, 14:16–23, Nov./Dec. 2000.